1. A shared input/output (I/O) fabric within a load/store domain, comprising:

   a plurality of root complexes;

   a shared I/O switch coupled to said plurality of root complexes; and

   a shared I/O controller, coupled to said shared I/O switch;

   wherein said shared I/O switch receives packets from each of said plurality of root complexes, places root complex identification within said packets for use by said shared I/O controller, and transmits said packets with said root complex identification to said shared I/O controller for processing.

2. The shared input/output (I/O) fabric as recited in claim 1 wherein the shared input/output (I/O) fabric utilizes a PCI Express Architecture.

3. The shared input/output (I/O) fabric as recited in claim 2 wherein said PCI Express Architecture is a base specification 1.0 that does not include provisions for sharing I/O.

4. The shared input/output (I/O) fabric as recited in claim 1 wherein said plurality of root complexes comprise a plurality of operating system domains (OSD's).

5. The shared input/output (I/O) fabric as recited in claim 4 wherein each of said plurality of OSD's has an operating system.

6. The shared input/output (I/O) fabric as recited in claim 5 wherein at least one of said plurality of OSD's executes an operating system that is different from other ones of said plurality of OSD's.

7. The shared input/output (I/O) fabric as recited in claim 4 wherein each of said plurality of OSD's comprise:

a processing complex; and

memory, coupled to said processing complex for storing data to be utilized by said processing complex.

8. The shared input/output (I/O) fabric as recited in claim 7 wherein said processing complex comprises one or more processors.

9. The shared input/output (I/O) fabric as recited in claim 1 wherein none of said plurality of root complexes include a dedicated network interface controller.

10. The shared input/output (I/O) fabric as recited in claim 1 wherein each of said plurality of root complexes comprise at least one port for coupling to said shared I/O switch.

11. The shared input/output (I/O) fabric as recited in claim 10 wherein said port conforms to a serial load/store architecture.

12. The shared input/output (I/O) fabric as recited in claim 10 wherein said port conforms to PCI Express architecture.

13. The shared input/output (I/O) fabric as recited in claim 1 wherein said shared I/O switch comprises a plurality of ports.

14. The shared input/output (I/O) fabric as recited in claim 13 wherein each of said plurality of root complexes is coupled to at least one of said plurality of ports to allow communication there between.

15. The shared input/output (I/O) fabric as recited in claim 1 wherein when said shared I/O switch receives a packet from one of said plurality of root complexes, it identifies which of said one of said plurality of root complexes transmitted said packet based on a port within said shared I/O switch coupled to said one of said plurality of root complexes.

16. The shared input/output (I/O) fabric as recited in claim 15 wherein the port is associated with a bus hierarchy within the load/store domain

17. The shared input/output (I/O) fabric as recited in claim 15 wherein said shared I/O switch places said root complex identification in said packet based upon said port.

18. The shared input/output (I/O) fabric as recited in claim 15 wherein said shared I/O switch associates said one of said plurality of root complexes that transmitted said packet with said one of said plurality of root complexes based on a PCI bus hierarchy within said shared I/O switch.

19. The shared input/output (I/O) fabric as recited in claim 15 wherein the association by said shared I/O switch utilizes a lookup table.

20. The shared input/output (I/O) fabric as recited in claim 1 wherein said shared I/O controller comprises:

a plurality of OS Domain portions.

21. The shared input/output (I/O) fabric as recited in claim 20 wherein each of said plurality of OS Domain portions comprise control registers.

22. The shared input/output (I/O) fabric as recited in claim 21 wherein each of said plurality of OS Domain portions further comprise a descriptor.

23. The shared input/output (I/O) fabric as recited in claim 1 wherein said shared I/O controller further comprises a bus interface for determining which of said plurality of root complexes is identified within said packets.

24. The shared input/output (I/O) fabric as recited in claim 1 wherein said shared I/O controller further comprises a bus interface for deterining which of a plurality of operating system domains (OSD's) is identified within said packets.

25. The shared input/output (I/O) fabric as recited in claim 1 wherein a bus interface within said shared I/O controller places said root complex identification within packets destined for said plurality of root complexes to allow said shared I/O switch to transmit said packets to an appropriate one of said plurality of root complexes.

26. The shared input/output (I/O) fabric as recited in claim 1 wherein said root complex identification comprises a plurality of bits which are inserted within said packets between said shared I/O switch and said shared I/O controller.

27. A serial communication architecture between a plurality of operating system domains (OSD's) and a plurality of endpoints to allow each of the plurality of OSD's to share each of the plurality of endpoints, the architecture comprising:

a first link, between each of the plurality of OSD's and a shared I/O switch;

a second link, between said shared I/O switch and each of the plurality of endpoints, said shared I/O switch associating packets from the plurality of OSD's with the OSD's by embedding a header within said packets before transmitting said packets to the plurality of endpoints.

28. The serial communication architecture as recited in claim 27 wherein the plurality of endpoints comprise:

a first endpoint for network communication; and

a second endpoint for communication with data storage devices.

29. The serial communication architecture as recited in claim 27 wherein the plurality of endpoints comprise:

a keyboard controller; and

a mouse controller.

30. The serial communication architecture as recited in claim 29 wherein the plurality of endpoints further comprise:

a video controller.

31. The serial communication architecture as recited in claim 28 wherein each of the plurality of root complexes communicates with a network through said first endpoint.

32. The serial communication architecture as recited in claim 27 wherein said first link comprises PCI Express.

33. The serial communication architecture as recited in claim 27 wherein said first link is replicated for each of the plurality of OSD's and said shared I/O switch.

34. The serial communication architecture as recited in claim 27 wherein said shared I/O switch comprises a plurality of ports, at least one of said ports associated with each one of the plurality of OSD's.

35. The serial communication architecture as recited in claim 27 wherein said second link comprises PCI Express+.

36. The serial communication architecture as recited in claim 27 wherein said second link comprises said first link plus an embedded field for storing said header.

37. The serial communication architecture as recited in claim 36 wherein said embedded field associates a packet from one of the plurality of OSD's with that one of the plurality of OSD's.

38. The serial communication architecture as recited in claim 36 wherein said embedded field comprises a six bit field for associating said packets with at least 64 distinct ones of the plurality of OSD's.

39. The serial communication architecture as recited in claim 36 wherein said embedded field associates said packets with ones of the plurality of OSD's utilizing a plurality of bit fields.

40. The serial communication architecture as recited in claim 27 wherein said header is transmitted with said packets from said shared I/O switch to at least one of the plurality of endpoints.

41. The serial communication architecture as recited in claim 40 wherein said at least one of the plurality of endpoints utilizes said header to determine which of the plurality of OSD's it is performing processing for.

42. The serial communication architecture as recited in claim 41 wherein each of the plurality of endpoints performs processing for at least two of the plurality of OSD's.

43. The serial communication architecture as recited in claim 41 wherein said packets from at least two of the plurality of OSD's reside within said at least one of the plurality of endpoints at the same time.

44. An apparatus for associating packets in a load/store serial communication fabric with root complexes to allow the root complexes to share an input/output (I/O) endpoint, the apparatus comprising:

a shared I/O switch, coupled to each of the root complexes, said shared I/O switch having routing control to associate the packets from each of the root complexes with the root complex they originate from by incorporating a field within the packets; and

a link between said shared I/O switch and the input/output (I/O) endpoint, wherein said link allows the packets to be transferred from said shared I/O switch to the input/output (I/O) endpoint with said field;

wherein the input/output (I/O) endpoint associates the packets with their associated root complexes by examining said field.

45. The apparatus for associating packets as recited in claim 44 wherein the packets are PCI Express packets.

46. The apparatus for associating packets as recited in claim 44 wherein the root complexes comprise:

a component in a PCI Express hierarchy that connects to a host bus segment on an upstream side with one or more PCI Express links on a downstream side.

47. The apparatus for associating packets as recited in claim 44 wherein the root complexes comprise operating system domains.

48. The apparatus for associating packets as recited in claim 44 wherein the root complexes comprise processing complexes.

49. The apparatus for associating packets as recited in claim 44 wherein the input/output (I/O) endpoint comprises a shared network interface controller.

50. The apparatus for associating packets as recited in claim 44 wherein the input/output (I/O) endpoint comprises a shared network storage controller.

51. The apparatus for associating packets as recited in claim 44 wherein the load/store serial communication fabric utilizes PCI Express.

52. The apparatus for associating packets as recited in claim 44 wherein said shared I/O switch comprises:

a plurality of ports for connecting said shared I/O switch to the root complexes and to the input/output (I/O) endpoint, wherein at least one of the plurality of ports is associated with each of the root complexes, and at least one of the plurality of ports is associated with the input/output (I/O) endpoint; and

said routing control, coupled to said plurality of ports, wherein said routing control is aware of which of said plurality of ports is associated with which of the root complexes.

53. The apparatus for associating packets as recited in claim 52 wherein said routing control comprises a table lookup that associates each of the plurality of ports with the root complexes.

54. The apparatus for associating packets as recited in claim 52 wherein said field includes information from said table lookup to associate the packets with their root complexes.

55. The apparatus for associating packets as recited in claim 44 wherein said link comprises a PCI Express+ link.

56. The apparatus for associating packets as recited in claim 44 wherein the input/output (I/O) endpoint contains packets from more than one of the root complexes at the same time.

57. A method for associating packets, within a serial load/store fabric, from a plurality of root complexes with their originating root complex, to allow the plurality of root complexes to share an I/O endpoint, the method comprising:

providing a first link between the plurality of root complexes and a switch, the packets in the first link unaware that the root complexes are sharing the I/O endpoint;

within the switch, embedding a header in the packets to associate the packets with their originating root complex;

providing a second link between the switch and the I/O endpoint, the second link capable of communicating the packets with the embedded header between the switch and the I/O endpoint; and

at the I/O endpoint, examining the packets with the embedded header to allow the I/O endpoint to associate each of the packets with their originating root complex.

58. The method for associating packets as recited in claim 57 wherein a first one of the root complexes is a network computer server with a Windows operating system.

59. The method for associating packets as recited in claim 58 wherein a second one of the root complexes is a network computer server with a Linux operating system.

60. The method for associating packets as recited in claim 57 wherein a first one of the root complexes comprises a processing complex.

61. The method for associating packets as recited in claim 60 wherein said processing complex comprises:

one or more processors; and

memory, coupled to said one or more processors for storing data utilized by said one or more processors.

62. The method for associating packets as recited in claim 57 wherein a first one of the root complexes comprises an operating system domain.

63. The method for associating packets as recited in claim 57 wherein the I/O endpoint is a 1 Gig Ethernet controller.

64. The method for associating packets as recited in claim 57 wherein the I/O endpoint is a 10 Gig Ethernet controller.

65. The method for associating packets as recited in claim 57 wherein the I/O endpoint is a Fiber Channel controller.

66. The method for associating packets as recited in claim 57 wherein the I/O endpoint is a serial ATA controller.

67. The method for associating packets as recited in claim 57 wherein the first link comprises a PCI Express link.

68. The method for associating packets as recited in claim 57 wherein the second link comprises a PCI Express+ link.

69. The method for associating packets as recited in claim 57 wherein said step of embedding comprises:

determining which port within the switch a packet is received from;

associating the packet with that port; and

assigning a header number to the packet associating the packet with a bus hierarchy for the port which received the packet.

70. The method for associating packets as recited in claim 69 wherein said step of associating the packet comprises:

performing a table lookup from a table which correlates that port with its associated root complex.

71. The method for associating packets as recited in claim 57 wherein a plurality of packets with embedded headers from a plurality of root complexes reside in the I/O endpoint at the same time.

72. A method for partitioning I/O devices among a plurality of processing complexes, the partitioning performed within the load/store domain of each of the processing complexes, the method comprising:

providing a switch between the I/O devices and the plurality of processing complexes, the switch utilizing a first load/store fabric between the plurality of processing complexes and the switch, and a second load/store fabric between the switch and the I/O devices;

mapping each of the plurality of processing complexes to one or more of the I/O devices; and

transferring packets between the plurality of processing complexes to the I/O devices based upon said mapping;

wherein at least one of the I/O devices is mapped to only one of the plurality of processing complexes; and

wherein at least one of the I/O devices is mapped to two or more of the plurality of processing complexes.

73. The method for partitioning I/O devices as recited in claim 72 wherein the switch is a shared I/O switch.

74. The method for partitioning I/O devices as recited in claim 72 wherein the first load/store fabric comprises PCI Express.

75. The method for partitioning I/O devices as recited in claim 72 wherein the switch of said step of providing utilizes a second load/store fabric between the switch and the I/O devices that includes header information associating packets with their processing complexes.

76. The method for partitioning I/O devices as recited in claim 75 wherein the second load/store fabric comprises PCI Express+.